WHAT IS CLAIMED IS:

1. A method of handling register spills in a parallel register architecture, comprising:

(i) determining whether register spill instructions in spill code generated by a register allocator can be associated;

(ii) if said register spill instructions can be associated, then rewriting said register spill instructions as a parallel register spill instruction;

(iii) based on said rewritten parallel register spill instruction, configuring storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel.

2. The method of claim 1, wherein said parallel architecture comprises a primary register set and a secondary register set, and (i) comprises determining whether two register spill instructions can be paired.

3. The method of claim 2, wherein (i) further comprises determining whether said two register spill instructions are in a basic block within said spill code.

4. The method of claim 3, wherein (i) further comprises determining whether said two register spill instructions relate to matching register locations in each of said primary register set and said secondary register set.

5. The method of claim 4, wherein (i) further comprises determining whether any intervening instructions between said register spill instructions modify either of said register spill instructions.

6. The method of claim 2, wherein (iii) comprises first allocating space on a memory stack to all paired register spills, then allocating space on said memory stack for any remaining register spills.

7. The method of claim 2, wherein (iii) comprises allocating space on said memory stack such that paired register spills are double word aligned.

8. The method of claim 7, further comprising loading said paired register spills from said memory stack back into matching register locations in each of said primary register set and said secondary register set in parallel.

9. A system for handling register spills in a parallel register architecture, comprising:

(a) a module for analyzing spill code generated by a register allocator to determine whether register spill instructions can be associated;

(b) a module for rewriting said register spill instructions as a parallel register spill instruction, if said register spill instructions can be associated;

(c) a module for configuring storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel based on said rewritten parallel register spill instruction.

10. The system of claim 9, wherein said parallel architecture comprises a primary register set and a secondary register set, and said module in (a) is configured to determine whether two register spill instructions can be paired.

11. The system of claim 10, wherein said module in (a) is further configured to determine whether said two register spill instructions relate to matching register locations in each of said primary register set and said secondary register set.

12. The system of claim 11, wherein said module in (a) is further configured to determine whether any intervening instructions between said register spill instructions modify either of said register spill instructions.

13. The system of claim 12, wherein said module in (c) is configured to first allocate space on a memory stack to all paired register spills, then allocate space on said

memory stack for any remaining register spills.

14. The system of claim 12, wherein said module in (c) is configured to allocate space on said memory stack such that paired register spills are double word aligned.

15. The system of claim 14, further comprising loading said paired register spills from said memory stack back into matching register locations in each of said primary register set and said secondary register set in parallel.

16. A system for handling register spills in a parallel register architecture, comprising:

(a) means for determining whether register spill instructions in spill code generated by a register allocator can be associated;

(b) means for determining if said register spill instructions can be associated, then rewriting said register spill instructions as a parallel register spill instruction;

(c) means for configuring, based on said rewritten parallel register spill instruction, storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel.

17. The system of claim 16, wherein said parallel architecture comprises a primary register set and a secondary register set, and (a) comprises means for determining whether two register spill instructions can be paired.

18. The system of claim 17, wherein (c) comprises means for allocating space on said memory stack such that paired register spills are double word aligned.

19. The system of claim 18, further comprising means for loading said paired register spills from said memory stack back into matching register locations in each of said primary register set and said secondary register set in parallel.

20. A computer readable medium having computer readable program code

embedded in the medium for handling register spills in a parallel register architecture, the computer readable program code including:

(i) code for determining whether register spill instructions in spill code generated by a register allocator can be associated;

(ii) code for determining if said register spill instructions can be associated, then rewriting said register spill instructions as a parallel register spill instruction;

(iii) code for configuring, based on said rewritten parallel register spill instruction, storage of associated register spills in memory in such a manner that said register spills can be loaded back into said registers in parallel.

21. The computer useable medium of claim 20, wherein said parallel architecture comprises a primary register set and a secondary register set, and (i) comprises code for determining whether two register spill instructions can be paired.

22. The computer useable medium of claim 21, wherein (iii) comprises code for allocating space on said memory stack such that paired register spills are double word aligned.

23. The computer useable medium of claim 22, further comprising code for loading said paired register spills from said memory stack back into matching register locations in each of said primary register set and said secondary register set in parallel.